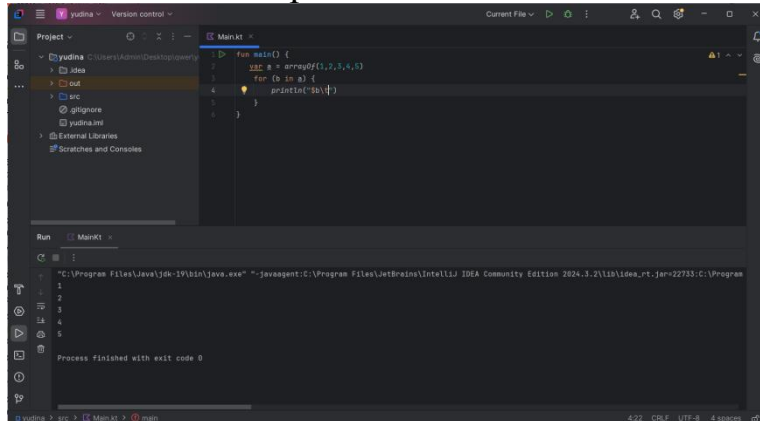


Практическая работа 6

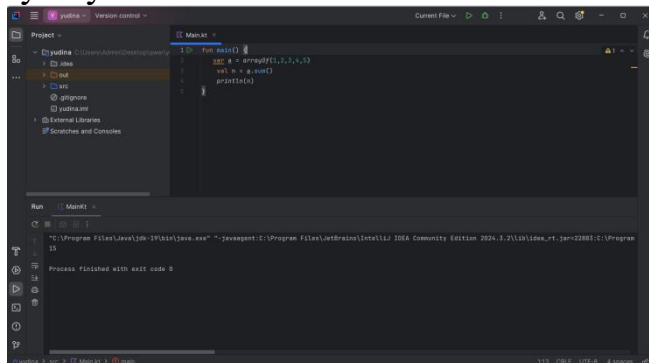
Массивы

1. Создание и вывод элементов: Создайте массив из 5 целых чисел и выведите их на экран.



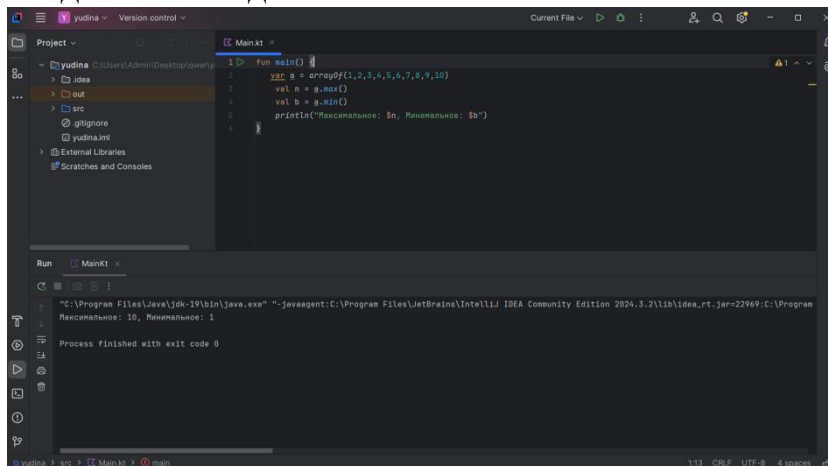
```
fun main() {  
    var a = arrayOf(1,2,3,4,5)  
    for (b in a) {  
        println("$b\t")  
    }  
}
```

2. Сумма элементов массива: Напишите программу, которая находит сумму всех элементов массива чисел.



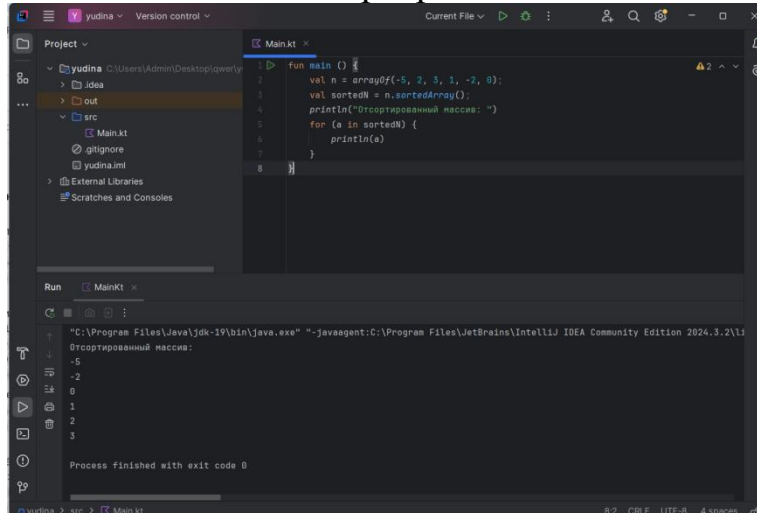
```
fun main() {  
    var a = arrayOf(1,2,3,4,5)  
    val n = a.sum()  
    println(n)  
}
```

3. Максимальное и минимальное значение: Создайте массив из 10 чисел, найдите и выведите максимальное и минимальное значение.



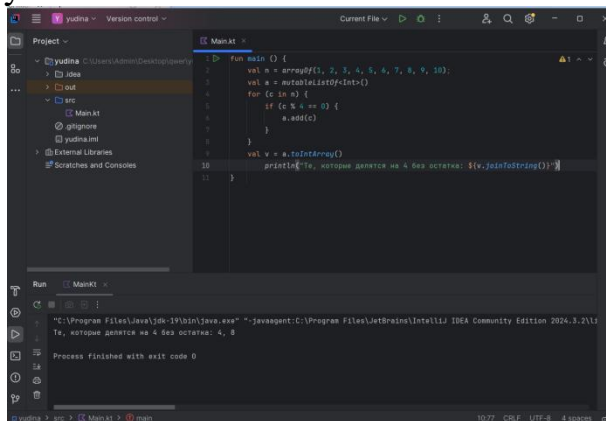
```
fun main() {
    var a = arrayOf(1,2,3,4,5,6,7,8,9,10)
    val n = a.max()
    val b = a.min()
    println("Максимальное: $n, Минимальное: $b")
}
```

4. Сортировка массива: Реализуйте алгоритм сортировки для массива чисел и выведите отсортированный массив.



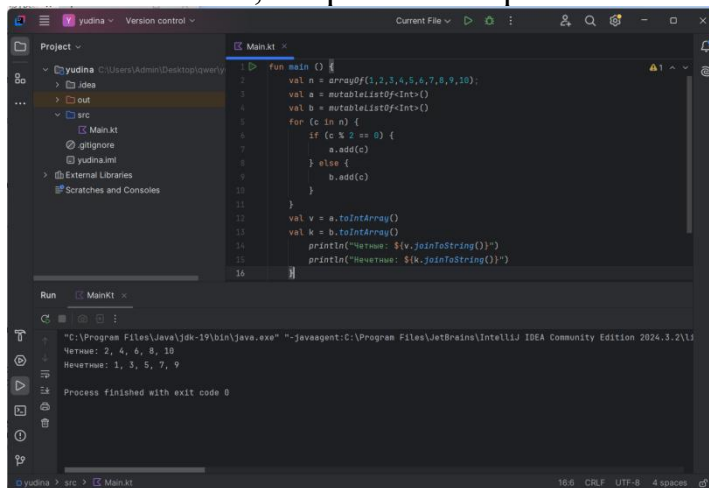
```
fun main () {
    val n = arrayOf(-5, 2, 3, 1, -2, 0);
    val sortedN = n.sortedArray();
    println("Отсортированный массив: ")
    for (a in sortedN) {
        println(a)
    }
}
```

5. Уникальные элементы: Напишите программу, которая выводит уникальные элементы из массива.



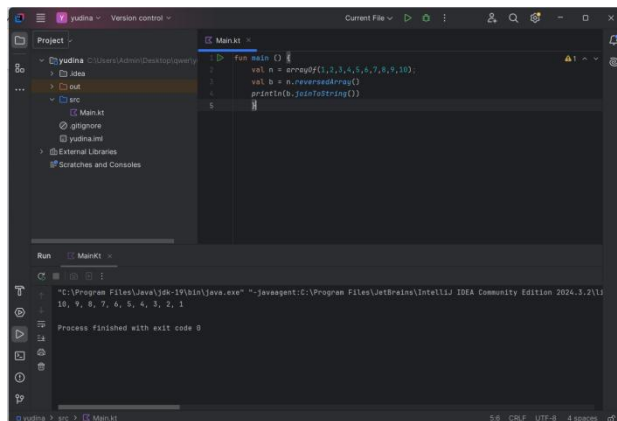
```
fun main () {
    val n = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
    val a = mutableListOf<Int>()
    for (c in n) {
        if (c % 4 == 0) {
            a.add(c)
        }
    }
    val v = a.toIntArray()
    println("Те, которые делятся на 4 без остатка: ${v.joinToString()}")
}
```

6. Четные и нечетные числа: Создайте массив и разделите его на четные и нечетные числа, сохранив их в разные массивы.



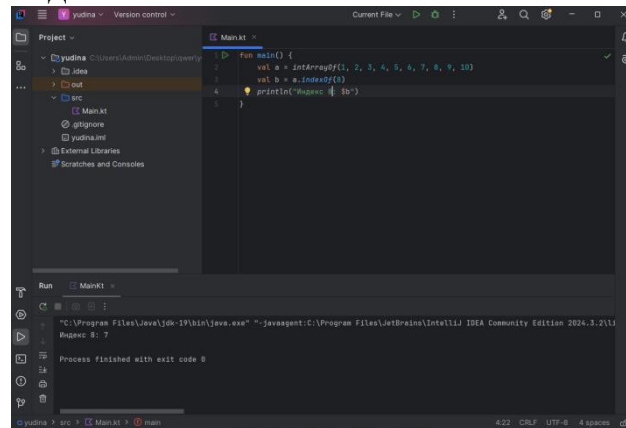
```
fun main () {
    val n = arrayOf(1,2,3,4,5,6,7,8,9,10);
    val a = mutableListOf<Int>()
    val b = mutableListOf<Int>()
    for (c in n) {
        if (c % 2 == 0) {
            a.add(c)
        } else {
            b.add(c)
        }
    }
    val v = a.toIntArray()
    val k = b.toIntArray()
    println("Четные: ${v.joinToString()}")
    println("Нечетные: ${k.joinToString()}")
}
```

7. Реверс массива: Напишите программу, которая реверсирует массив чисел.



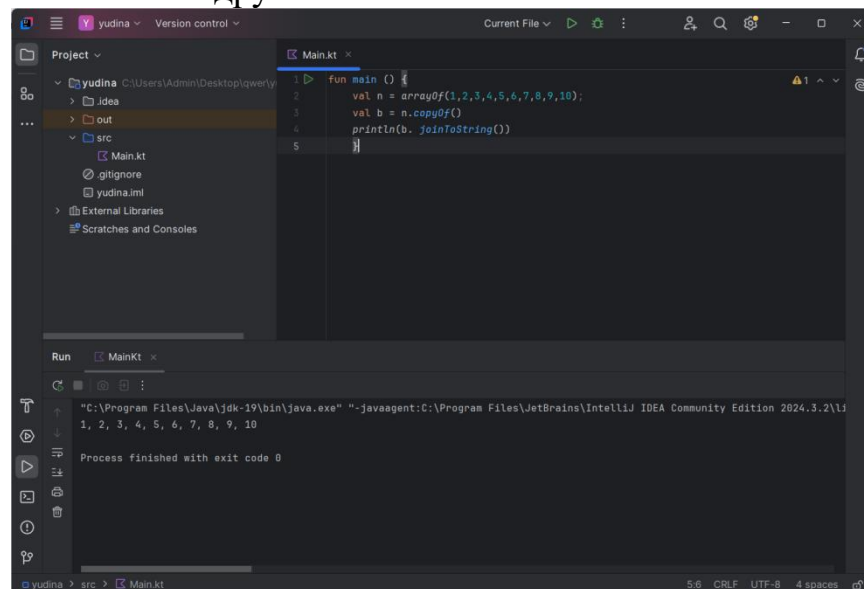
```
fun main () {
    val n = arrayOf(1,2,3,4,5,6,7,8,9,10);
    val b = n.reversedArray()
    println(b.joinToString())
}
```

8. Поиск элемента: Реализуйте поиск элемента в массиве и выводите его индекс.



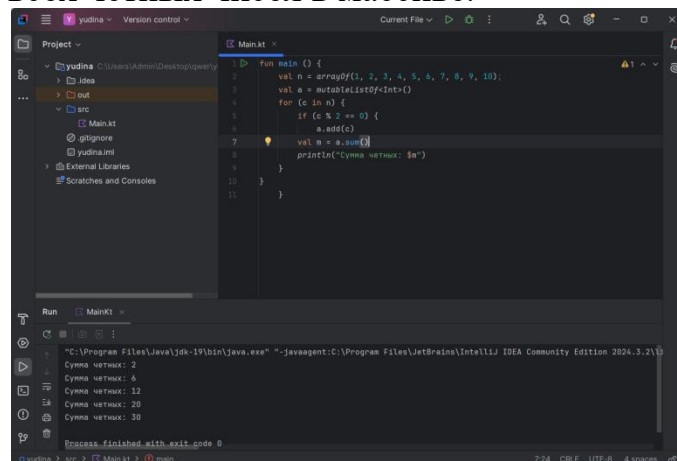
```
fun main() {  
    val a = intArrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)  
    val b = a.indexOf(8)  
    println("Индекс 8: $b")  
}
```

9. Копирование массива: Создайте новый массив, скопировав в него элементы из другого массива.



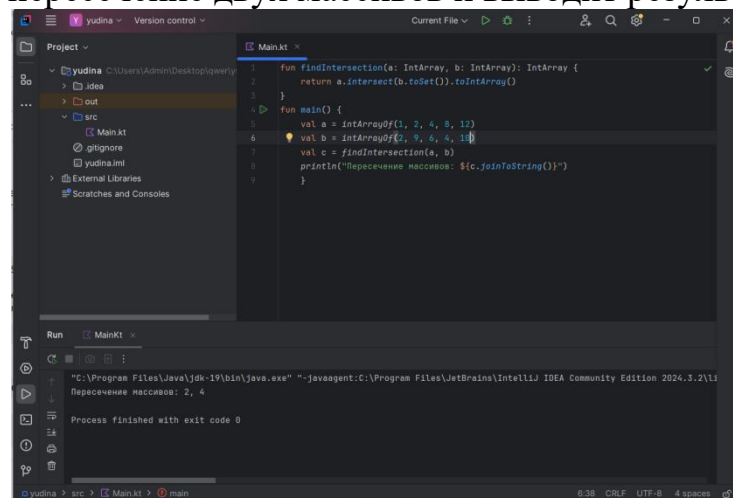
```
fun main () {  
    val n = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);  
    val b = n.copyOf()  
    println(b.joinToString())  
}
```

10. Сумма четных чисел: Напишите программу, которая находит сумму всех четных чисел в массиве.



```
fun main () {
    val n = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
    val a = mutableListOf<Int>()
    for (c in n) {
        if (c % 2 == 0) {
            a.add(c)
        }
        val m = a.sum()
        println("Сумма четных: $m")
    }
}
```

11. Пересечение массивов: Напишите программу, которая находит пересечение двух массивов и выводит результат.

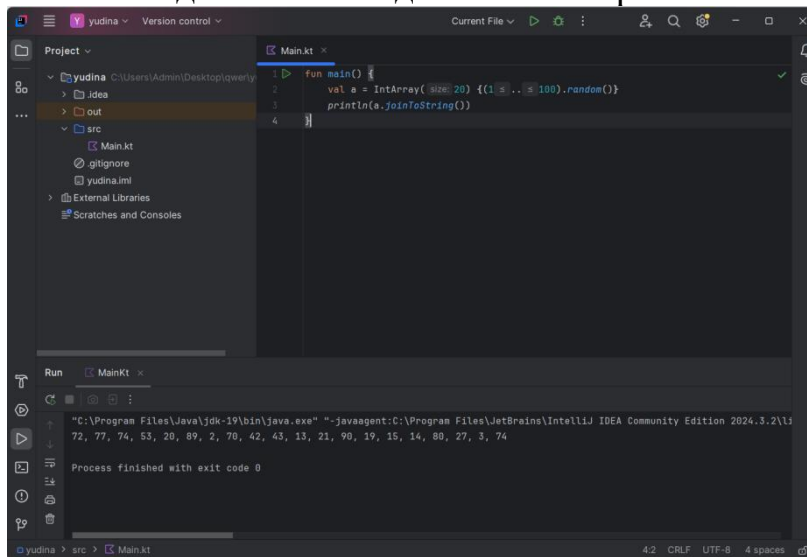


```
fun findIntersection(a: IntArray, b: IntArray): IntArray {
    return a.intersect(b.toSet()).toIntArray()
}

fun main() {
    val a = intArrayOf(1, 2, 4, 8, 12)
    val b = intArrayOf(2, 9, 6, 4, 10)
    val c = findIntersection(a, b)
    println("Пересечение массивов: ${c.joinToString()}")
}
```

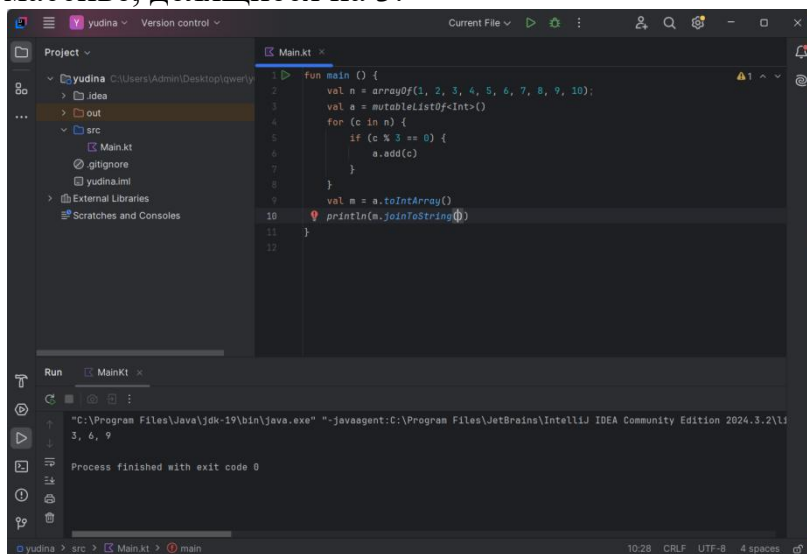
12. Перестановка элементов: Реализуйте функцию, которая меняет местами два элемента в массиве.

13. Заполнение случайными числами: Создайте массив из 20 случайных чисел от 1 до 100 и выведите его на экран.



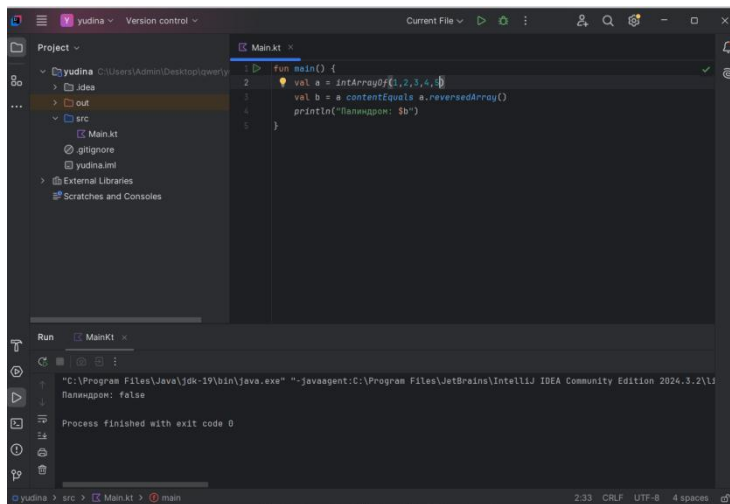
```
fun main() {  
    val a = IntArray(20) { (1..100).random() }  
    println(a.joinToString())  
}
```

14. Числа Прокопенко: Напишите программу, которая выводит все числа в массиве, делящиеся на 3.



```
fun main () {  
    val n = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);  
    val a = mutableListOf<Int>()  
    for (c in n) {  
        if (c % 3 == 0) {  
            a.add(c)  
        }  
    }  
    val m = a.toIntArray()  
    println(m.joinToString())  
}
```

15. Проверка на палиндром: Напишите программу, которая проверяет, является ли массив палиндромом.

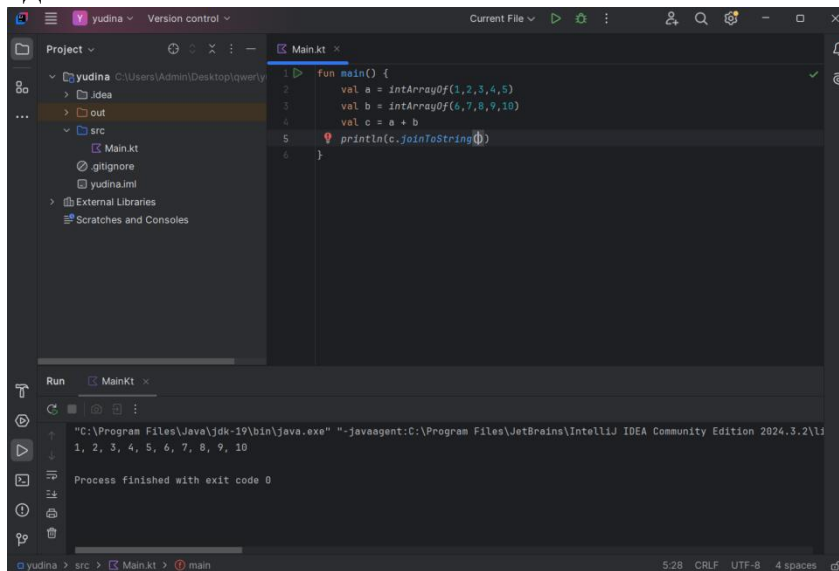


```

fun main() {
    val a = intArrayOf(1, 2, 3, 4, 5)
    val b = a.contentEquals(a.reversedArray())
    println("Палиндром: $b")
}

```

16. Конкатенация двух массивов: Создайте два массива и соедините их в один.



```

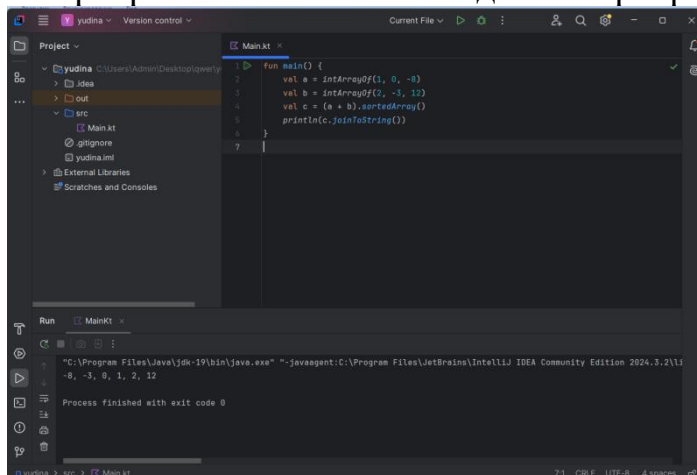
fun main() {
    val a = intArrayOf(1, 2, 3, 4, 5)
    val b = intArrayOf(6, 7, 8, 9, 10)
    val c = a + b
    println(c.joinToString())
}

```

17. Сумма и произведение: Напишите программу, которая находит и выводит сумму и произведение всех элементов массива.

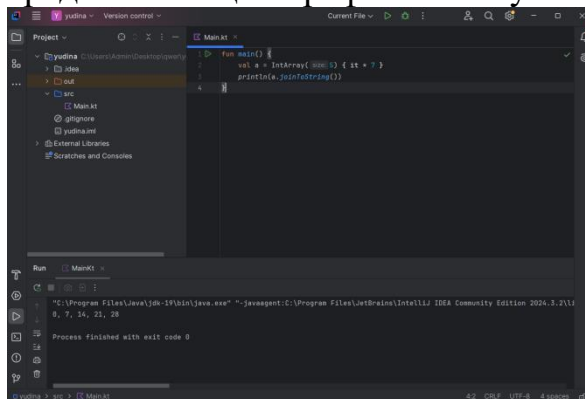
18. Группировка чисел: Разделите массив на группы по 5 элементов и выведите их.

19.Слияние двух массивов: Напишите программу, которая сливает два отсортированных массива в один отсортированный массив.



```
fun main() {  
    val a = intArrayOf(1, 0, -8)  
    val b = intArrayOf(2, -3, 12)  
    val c = (a + b).sortedArray()  
    println(c.joinToString())  
}
```

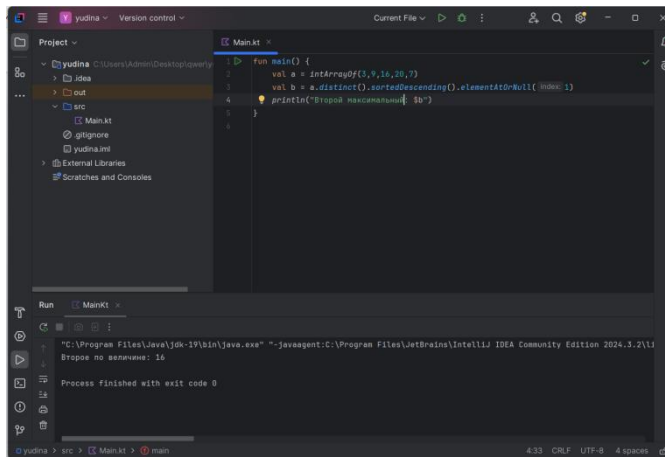
20.Числовая последовательность: Создайте массив целых чисел, представляющий арифметическую прогрессию, и выведите его.



```
fun main() {  
    val a = IntArray(5) { it * 7 }  
    println(a.joinToString())  
}
```

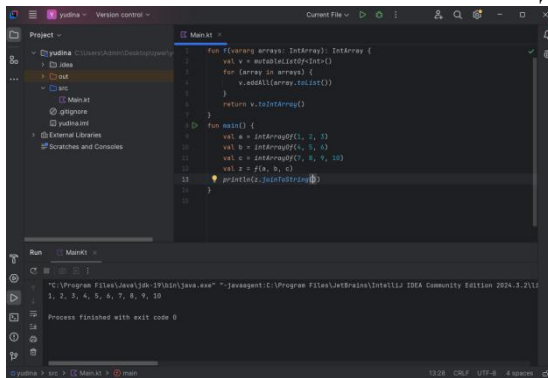
21.Удаление элемента: Реализуйте функцию, которая удаляет заданный элемент из массива.

22.Поиск второго максимального: Напишите программу, которая находит второй по величине элемент в массиве.



```
fun main() {
    val a = arrayOf(3, 9, 16, 20, 7)
    val b = a.distinct().sortedDescending().elementOrNull(1)
    println("Второй максимальный: $b")
}
```

23. Объединение массивов: Напишите функцию, которая объединяет несколько массивов чисел и выводит результирующий массив.

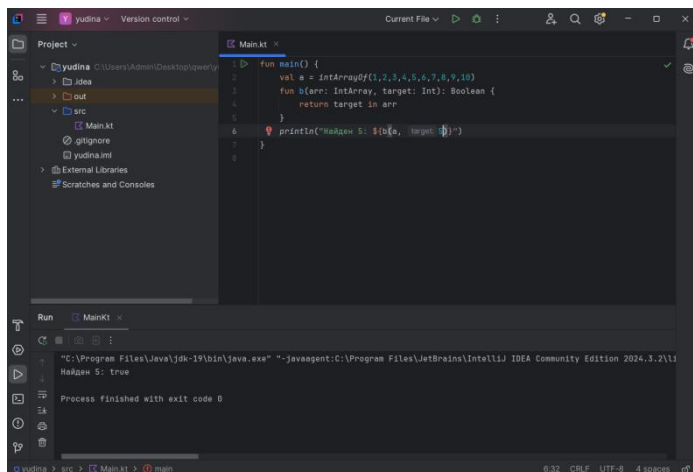


```
fun f(vararg arrays: IntArray): IntArray {
    val v = mutableListOf<Int>()
    for (array in arrays) {
        v.addAll(array.toList())
    }
    return v.toIntArray()
}

fun main() {
    val a = arrayOf(1, 2, 3)
    val b = arrayOf(4, 5, 6)
    val c = arrayOf(7, 8, 9, 10)
    val z = f(a, b, c)
    println(z.joinToString())
}
```

24. Транспонирование матрицы: Создайте матрицу (двумерный массив) и напишите программу, которая транспонирует её.

25. Линейный поиск: Реализуйте линейный поиск элемента в массиве с Возвратом Bool-значения (найден или нет).

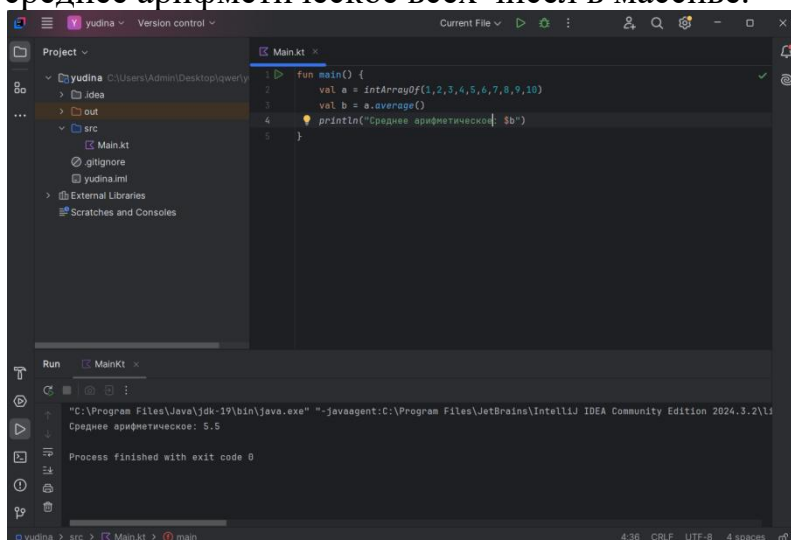


```

fun main() {
    val a = intArrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
    fun b(arr: IntArray, target: Int): Boolean {
        return target in arr
    }
    println("Найден 5: ${b(a, 5)}")
}

```

26. Среднее арифметическое: Напишите программу, которая находит среднее арифметическое всех чисел в массиве.



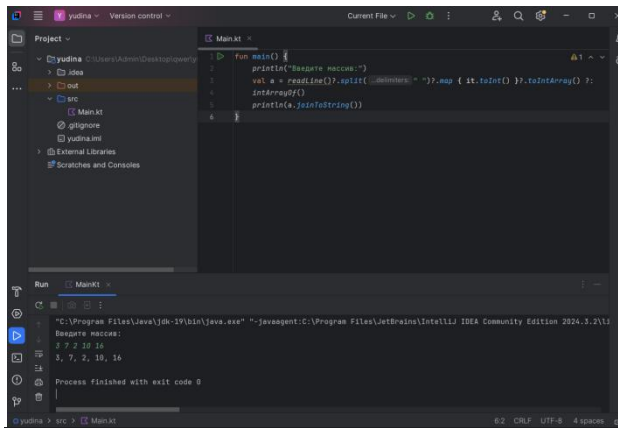
```

fun main() {
    val a = intArrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
    val b = a.average()
    println("Среднее арифметическое: $b")
}

```

27. Максимальная последовательность: Найдите максимальную последовательность одинаковых элементов в массиве.

28. Ввод и вывод массива: Напишите программу, которая запрашивает у пользователя ввод массива чисел и затем выводит его.



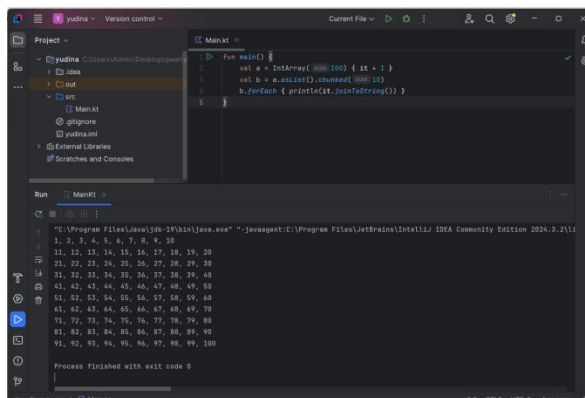
```

fun main() {
    println("Введите массив:")
    val a = readLine()?.split(" ").map { it.toInt() }.toIntArray() ?:
    intArrayOf()
    println(a.joinToString())
}

```

29.Нахождение медианы: Напишите программу, которая находит медиану в массиве.

30.Распределение по группам: Создайте массив из 100 целых чисел и разделите их на 10 групп по 10 элементов, затем выведите результаты.



```

fun main() {
    val a = IntArray(100) { it + 1 }
    val b = a.asList().chunked(10)
    b.forEach { println(it.joinToString()) }
}

```

